# 一种解决扑克博弈的新思路

## 介绍

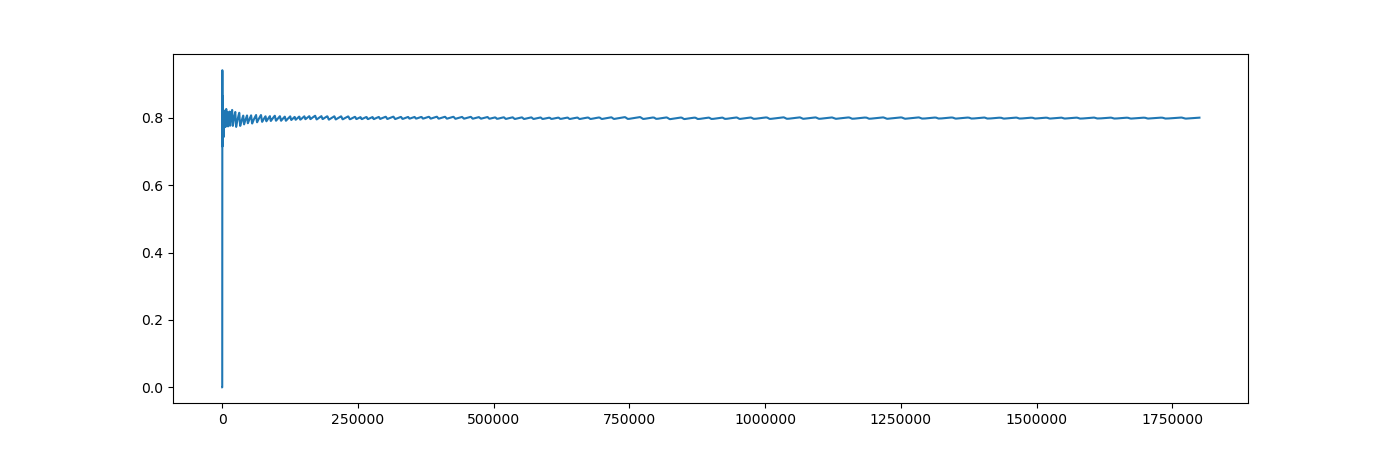
目前大多数人工智能博弈研究，都会遵循一个基本规则，即人工智能要和人类站在同一起跑线上。比如在星际中，人工智能的输入也是屏幕，人工智能的每分钟操作数不能超过人类选手。在德州扑克中，人工智能比赛时，也不能知道对手的手牌。

但是，如果我们的目标仅仅设为找到博弈的均衡。那么假如人工智能在训练的过程中，能够知道对手的策略，手牌等信息，是否会使得收敛到均衡的速度增加，并没有太多这方面的研究。

目前计算均衡最先进的算法就是CFR，但是在CFR过程中，很多时候，受随机的影响非常大。由于存在随机性，这使得玩家找到的后悔值可能和真实后悔值差距很大，但是倘若玩家知道对手的策略，那么玩家就可以做出更加精确的判断。譬如一个出硬币猜正反的游戏，如果出硬币者出正面的概率为51%反面49%，在CFR过程中对于猜测者来说，需要非常多轮的观察计算后悔值，才能得出对方出正面比出反面概率高的结论，并且这个结论有概率会是错的；但是倘若出硬币者直接把其策略告知猜测者，那么猜测者马上可以做出正确应对，这从理论上讲，确实会加快收敛速度。

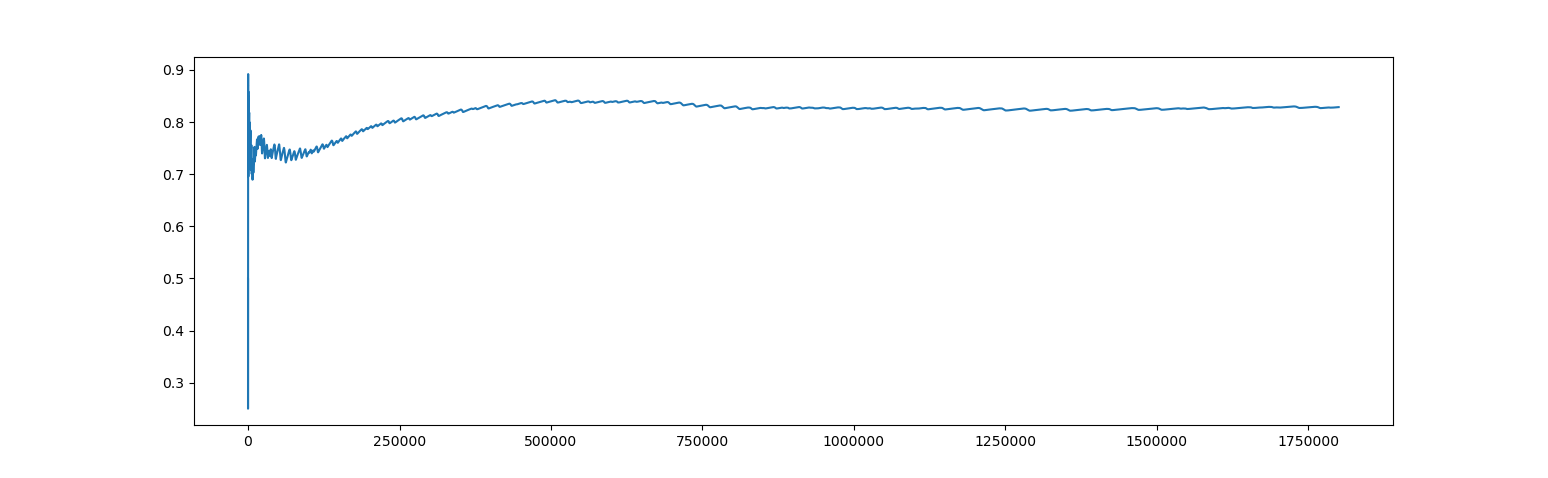
### CFR收敛情况

但是情况也要分成两种来考虑，目前我的测试结果(可能是我没理解透CFR)是，如果均衡（这里及以后的均衡都是指混合均衡）是单一的，CFR则会收敛的很快。但是如果存在多均衡，则CFR收敛会大大减慢。但是下面展示着两种情形，以下的博弈均基于kuhn扑克。图一为如果两方都赌，则赢家得3元，其余情况赢家得2元（这种情况只有一种均衡）。图二为原始kuhn扑克，如果两方都赌，则赢家得2元，其余情况赢家得1元（这种情况存在无穷多种均衡）。



上图博弈仅存在单一混合均衡，最后波动收敛。约收敛到小数点后4位。

[0.800731 0.999996 0.000002]

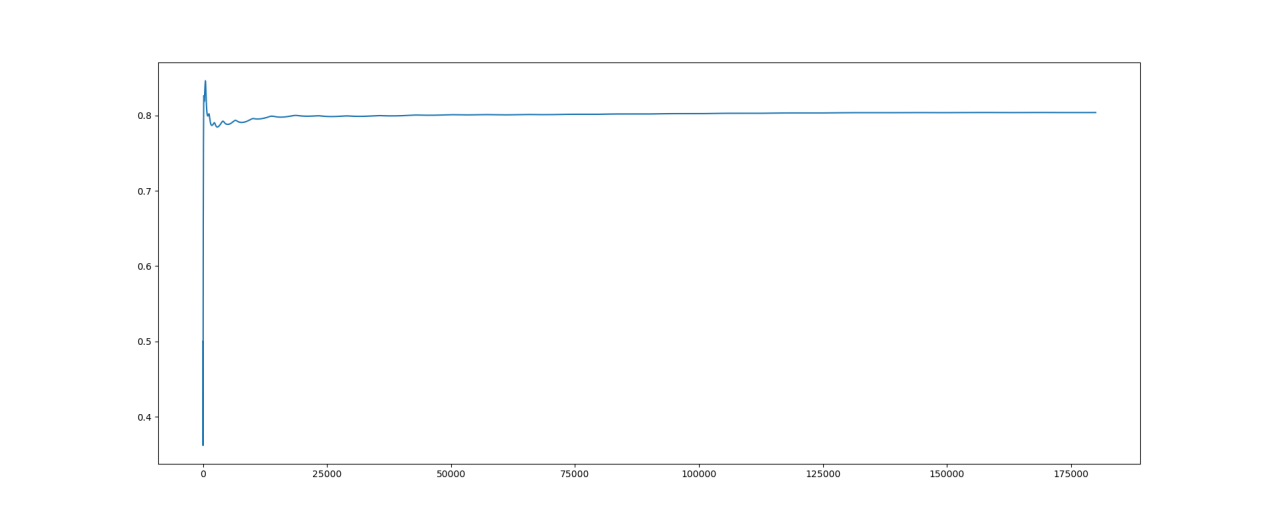


上图是如图所示CFR会收敛。但是倘若存在混合均衡，CFR的效率会大大缩减可以看到开始时波动很大，而且仅收敛到约小数点后3位。

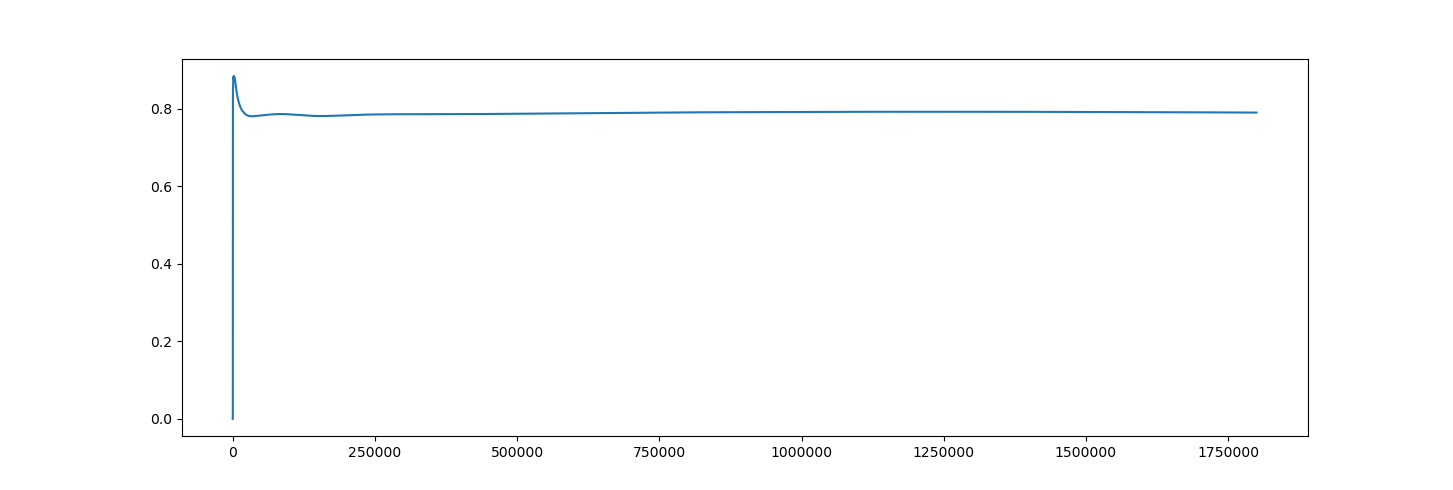
[0.678328 0.999997 0.037672]

[0.999999 0.344713 0.000011]

要注意的是在只存在单一均衡的情况下，还有更快的收敛办法，只要求前面累计策略平均，收敛就会加快非常多。



但是要注意，如果不是单一均衡而求平均会导致结果偏离均衡(此时虽然收敛加快，但是不是均衡)。



## 扩展式博弈，策略，纳什均衡，遗憾匹配

### 定义1 [8, p.200]具有不完全信息的有限广义博弈具有以下成分：

* 确定的N个玩家
* 序列的有限集H，可能发生的动作的历史记录，以使空序列位于H中，而序列中每个H的前缀也位于H中。是终点状态。代表序列H后可采取的动作。
* 函数P确定每个非终止节点上的可以行动的人.如果是P(h)=c则是随机事件。
* 函数联系历史h到所出的动作，其中如果P(h)=c就是一种概率度量，就是出现h是动作a出现的概率。其中每个此类概率测度均独立于其他所有此类测度。
* 对于每个玩家i∈N，{h∈H：P（h）= i}的信息集分区Ii具有以下属性：每当h和h’在分区的同一成员中时，A（h）= A（h’）。对于Ii∈Ii，我们用A（Ii）表示集合A（h），对于任何h∈Ii，用P（Ii）表示玩家P（h）。Ii是玩家i的信息分区； 集合Ii∈Ii是玩家i的信息集。（这里翻译不准确）
* 对于每个参与者i∈N，从终端状态Z到实数R的效用函数ui。如果N = {1，2}并且u1 = -u2，则它是零和广义博弈。将定义为玩家i的效用范围。

但是德州扑克有一种更优良的性质，即无论是我对于对手的不确定性，还是对手对我的不确定性，都只发生在游戏一开始，其后的所有动作，包括荷官发牌，都是可见的信息。于是针对扑克博弈我们加入以下定义：

* Poker\_num代表手牌可能的情况。
* 集合代表在已经发生的行动序列h上，玩家i手牌的概率组合。其中代表玩家i在发生h时手握第x种手牌的可能性。注意当h为初始状态时,；当h已经进行了y轮博弈后并且在博弈树完全的情况下。代表在已发生动作序列h上所有玩家手牌组合的联合概率分布。令。
* 代表在已发生行动序列h后的虚拟博弈。可以理解为，玩家无论拥有何种手牌时都一定遵循h进行游戏。此时，当h为空时代表整个博弈。注意其与子博弈的区别，即子博弈代表在到达行动序列h时会发生变化，但是虚拟博弈上，仍然是初始状态，不会变化。（这里可能需要讨论）
* 定义代表次动作序列包含行动的次数。集合代表，离最终点还有n步的距离。其中代表行动h之后再接一个行动序列h’。

此外，需要强调的是本工作将专注于具有完美回溯的有限零和广义博弈。

### 策略

博弈中玩家i，的策略是一项将A（Ii）上的分布分配给每个Ii∈Ii的函数，而Σi是玩家i的策略集。策略组合σ由每个参与者的策略组成。其中指的是σ中除σi之外的所有策略.

是如果按照策略出牌h 出现的概率。我们可以把分开，

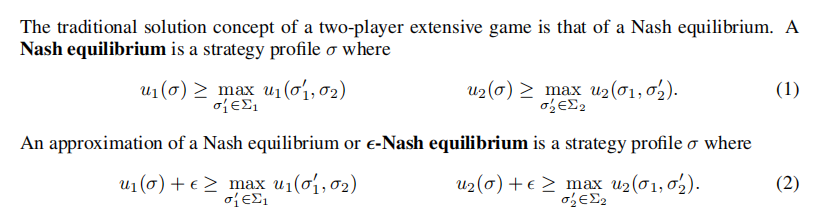
变成各个玩家对这个情况出现概率的贡献。因此，是玩家i根据σ进行游戏的概率，那么对于所有历史（即h的适当前缀，且），玩家i在h中采取相应的动作。令是除玩家i外的所有玩家贡献（包括机会）的乘积。对于I⊆H，定义，作为达到特定信息集σ的概率，其中和的定义类似。

然后，策略配置文件对玩家i的总价值就是最终终端节点的预期收益，。

同样针对德州扑克，我们定义：

* 代表，在策略下，玩家i把自己的手牌组合分配到h的后继可能组合中。
* 代表在虚拟博弈上使用策略时玩家i所得的收益。
* 定义代表除了在行动序列h上策略等于，其余行动序列不变。

### 纳什均衡

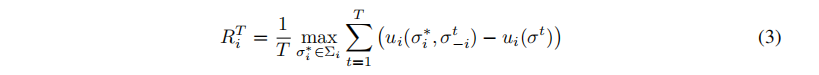


（这里还要进一步讨论，即子博弈精炼均衡和相关均衡）

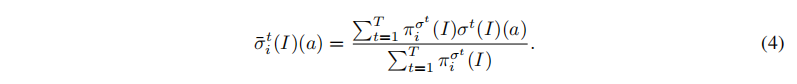
在非完全信息零和博弈中，均衡的收益只会有一种情况。而不会像非零和博弈会出现多个均衡，并且这些均衡之间收益不相等的情况。零和博弈中即使存在多均衡，他们的收益也会一样。

### 遗憾匹配

遗憾是一种在线学习概念，已触发了一系列强大的学习算法。为了定义这个概念，首先要考虑反复玩一个广泛的游戏。是玩家在t轮采取的策略。玩家i在时间T的平均总体遗憾为



另外，定义为玩家1从游戏次数1到T的平均策略。特别的，对每个信息集I定义。



这是一个众所周知的遗憾与纳什均衡解之间的联系。

### 定理2：在零和博弈中，如果玩家的整体遗憾小于则此策略是纳什均衡。

如果玩家i的平均总体遗憾（与序列t无关）随着t趋于无穷大而为零，那么为玩家i选择的算法就将遗憾最小化。 所以，在左右互搏中的遗憾最小化算法可以用作计算近似Nash均衡。此外，算法在平均总体遗憾程度上的界限限制了近似值的收敛速度。

## 新思路介绍(写的很菜还要再完善)

新思路的解决方式是把遗憾分别求解，首先对于两人博弈我们计算出虚拟博弈到终点Z的手牌联合概率分布，根据效用函数，计算在根节点上每一种联合分布的收益矩阵。此时在中我们可以找到最优策略，其中。我们已经假设了博弈是零和博弈，故策略一定是一个纯战略。定义代表在迭代第t轮，在行动序列h上，手持手牌x，选择动作a的后悔值。此时我们可以计算，。然后将也看为Z，其收益。在此基础上计算虚拟博弈，依次类推。最后根据依据后悔值算法计算出下一次策略。

## 评价介绍

由于我没有完成数学的推到，所以在这里，通过运行多次求平均值，然后进行数据拟合的方法推算出新思路和CFR的收敛情况并做比较。介绍新方法效率之前，首先要讨论如何评价新方法。

## 实验博弈介绍

下面介绍使用的测试博弈，测试博弈来源于kuhn扑克，但是做了两种改变。改变1是原kuhn扑克牌只有三张，现在可以任意扩展成10张，20张甚至更多张(修改桶内种类)；改变2是原本kuhn扑克只能进行一次加注跟住，现在变成最多可进行3次加注跟住(增加桶数)。以下测试如果是新思路是测试500轮取迭代1000次取平均。CFR则是测试300轮迭代1000\*(牌数)轮，因为根据原论文只要CFR迭代次数和博弈的信息集数量比例一定，其收敛速度。

### 算法效率之收敛性

对于算法来说最重要的是随着迭代次数增加，算法收敛到均衡的速度。所以我们需要一种度量，来描述一个策略离均衡的距离。

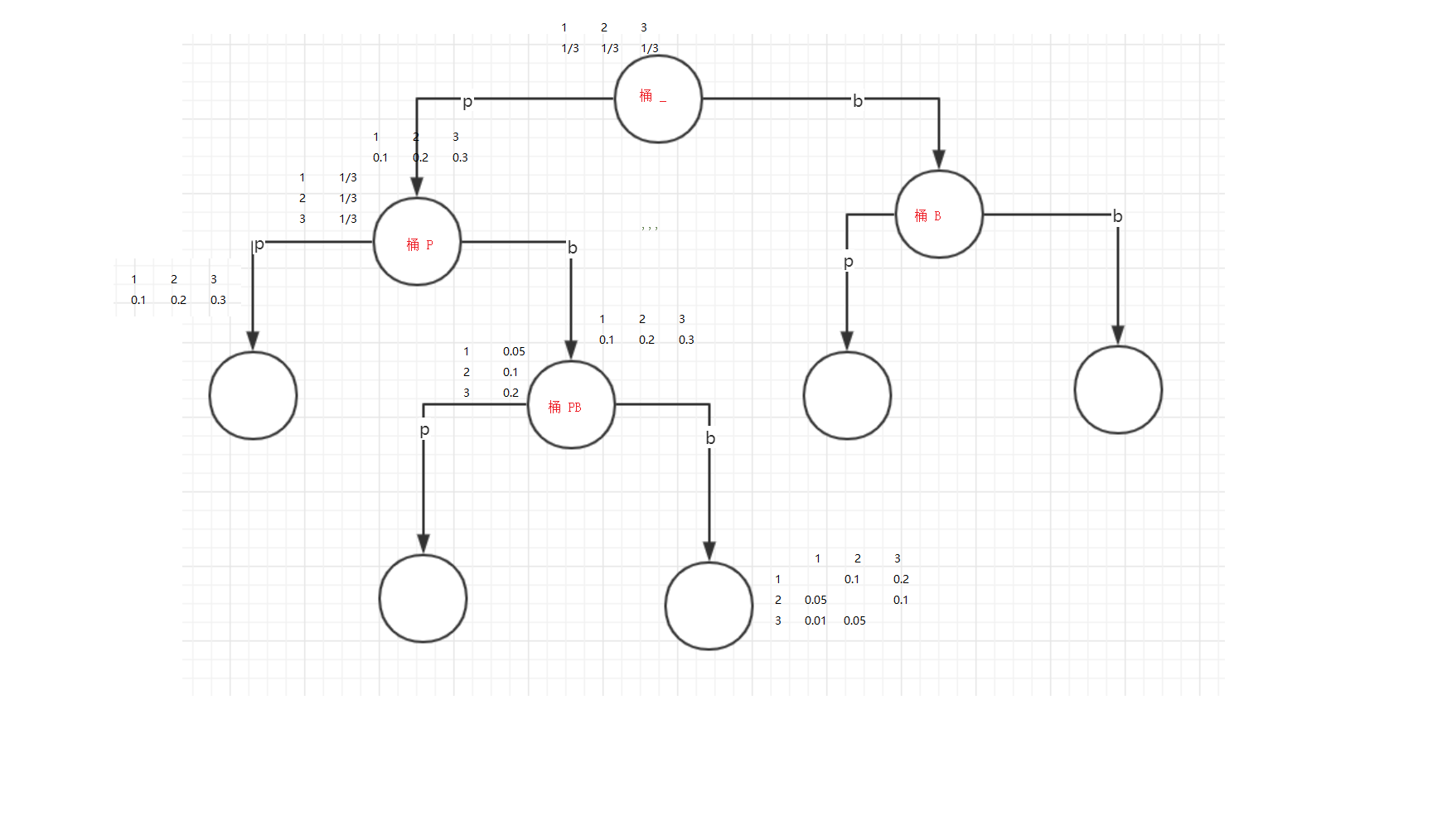
在这里，我们使用了子博弈精炼纳什均衡的思想。

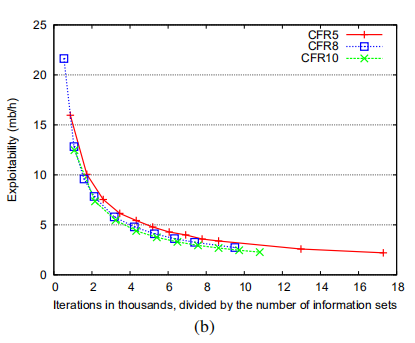
首先针对每一个虚拟博弈计算出，寻找在虚拟博弈上改变策略带来的最大收益,令代表，假如在虚拟博弈上改变行动序列h时的博弈策略能带来的收益提升。令代表整个博弈经过调整策略，能带来的收益提升之和(对于另一个玩家来说就是可利用度)。并把这个值，作为离真实均衡的差距的度量。

### 算法效率之迭代时间复杂度

1. 我们需要知道随着信息集的增加，算法的收敛速度如何变化。这其中涉及到了两种信息集，首先是“桶”(bucket)的数量，也就是h的可能情况；第二个是桶中分类的数量，也就是poker\_num的数量。

桶的意思是在给定不讨论手牌的情况有多少个决策节点，比如2人kuhn扑克，虽然有12个信息集，但是只有4个桶，分别是‘\_’，‘P’，‘PB’，‘B’。其次是桶中分类的数量，在kuhn扑克中可以理解成手牌种类数量。如原始的3张扑克牌随机发牌，或者我们扩展到5张扑克牌随机发牌等等。





## 测试结果

### 收敛性

下面是测试结果五张牌，两轮加注迭代1000次时可以看出收敛情况如下。

可以发现效率和迭代次数的负幂次成正比，这和CFR类似。但是收敛的比CFR快，同类型CFR如下(此CFR收敛不是论文里提及收敛速度，可能是我理解错了，也可能是计算可利用度方法不同造成的)。

### 改变牌数

现在改变牌数为3牌，7牌，9牌，观察随着桶内种类变化收敛速度的变化。

可以看出随着牌数变化，新思路的收敛速度是在变化的。但是CFR就有明显不同，可以看出CFR收敛速度几乎不变，这时和论文里提到的情况是一致的。

### 改变加注数

在只能进行一次加注的时候，新方法的时间确实在减小，但是我在测试时发现CFR的迭代没有像论文中说的一样不变（这个也是问题），可能是CFR在这时可能是由于循环次数不够多所以并不精准。但是在我认真核对之后在3次赌注时会出现难以收敛的情况，目前我找到两个可能的原因，一是因为迭代时如果出现两个策略的后悔值都是0，则CFR无法给出结果只能给出两个策略1/2,1/2导致；二是，有可能说明CFR并不是收敛到子博弈精炼纳什均衡。

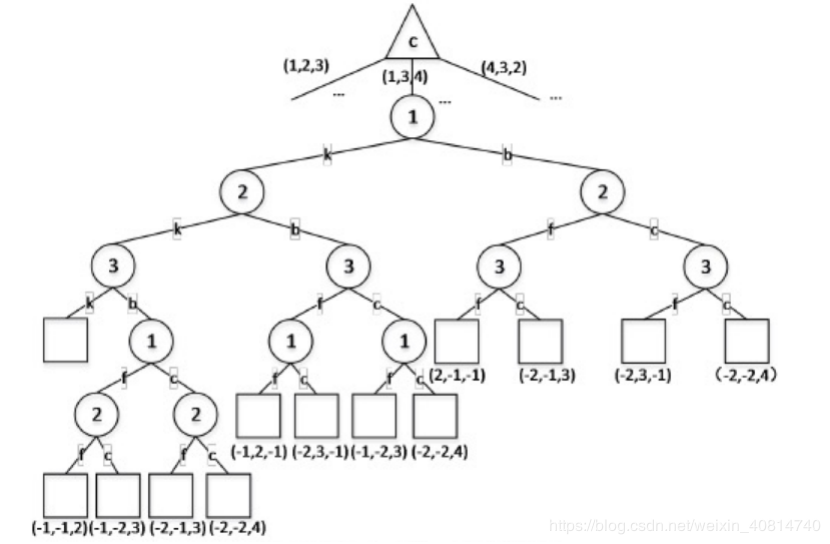
### 迭代时间

新方法和CFR随着桶数的增加耗时一样线性增加，但是新方法迭代时间随着牌数增加是二次关系，由于在计算过程中，是直接带着矩阵进行计算，矩阵运算时时间都是二次关系。但是CFR计算时并不是二次关系，因为只要桶的数量不变的情况下一次CFR的时间是不变的，所以CFR的时间随着迭代次数变化而线性变化。

## 对于新思路的猜想与讨论

在看了对比之后可能还无法准确理解这其中的差距。虽然看起来迭代次数相似的情况下收敛精度约只比CFR只快了大约10倍。但是由于随着迭代次数增加，收敛会越来越慢，在我这种误差评估思想下，CFR几乎迭代次数要多到新方法几千倍才能达到相似的收敛精度。举几个例子，比如面对一个5牌2加注的问题CFR经历5000次迭代的结果新思路只要迭代18次就能得到；如果是3牌3加注的库恩扑克，CFR经历了三万次收敛得到的精度，新思路只要惊人的6轮就达到了。虽然新思路的迭代时间随着bucket上升成二次上升，但是这个提升实在是太大了，所以我甚至非常怀疑是由于程序bug导致，但是我确实没找到哪里有问题。如果没有写错的。这也有可能是计算误差的方式导致的，计算误差的方式也需要进一步讨论。

## 附录A kuhn扑克玩法

下面我们介绍一种游戏kuhn扑克。Kuhn 扑克(Kuhn Poker)是最简单的限注扑克游戏。虽然最原始的规则仅仅需要两个玩家，但 2010 年 Abou Risk 和 Szafron创建了三人的版本，现在规则已经可以适用于 n 个玩家。Kuhn Poker 整付牌有 n+1 张，牌值为 1，2，...，n+1，1 最小，n+1 的牌值最大。每个玩家手持一张底牌和将一定底注投入彩(盲注)。Kuhn Poker 只有一轮且要在翻牌前进行押注，押注筹码是固定的，游戏不允许加注。游戏开始玩家可以选择过牌或者押注，如果一旦有人押注，则其他玩家(包括顺位在押注玩家之后的玩家，和押注玩家之前过牌的玩家)则只能选择弃牌或者跟注。游戏没有公共牌，摊牌阶段比较未弃牌的玩家中底牌牌值大小，底牌牌值最大的玩家即为获胜者。下图展示了 3 人 Kuhn 扑克扩展式博弈树的子博弈树。

图中显示为3个玩家手牌为1，3，4的情况。K ,b ,f , c分别代表，过牌，押注，弃牌，跟注。图中盲注大小为1，押注大小也为1。两人kuhn扑克博弈收益如下图。

